Лучшее малоранговое приближение: теорема Эккарта–Янга

Самое замечательное свойство SVD состоит в том, что оно позволяет *ужимать* матрицу, выбрасывая всё «несущественное».

Разложение по слагаемым ранга один

Перепишем (0.1) по-другому. Обозначим столбцы матрицы U через $u_1, \ldots, u_m$ (это *левые сингулярные векторы*), а столбцы V — через $v_1, \ldots, v_n$ (*правые сингулярные векторы*). Тогда A есть сумма r слагаемых, каждое из которых имеет ранг один:

$$A = \sum_{i=1}^{r} \sigma_i \, u_i v_i^{\top}. \tag{1}$$

Каждое слагаемое $u_i v_i^{\top}$ — это «вертикальный вектор умножить на горизонтальный», и оно само по себе является матрицей ранга один (рис. 0.2).

> ⚠ Рисунок
>
> Рисунок fig:rank1 не сгенерирован: C:\Users\712\AppData\Local\MiKTeX\miktex\log\miktex-maketfm.log

Рис. 0.2. Матрица ранга один: каждый столбец результата пропорционален u, каждая строка — пропорциональна $v^{\top}$. Чтобы её хранить, достаточно $m + n$ чисел вместо mn.

Если оборвать сумму (0.3) на k-м слагаемом ($k < r$), получим приближение:

$$A_k = \sum_{i=1}^{k} \sigma_i \, u_i v_i^{\top}. \tag{2}$$

Это матрица того же размера $m \times n$, но ранга всего лишь k.

Теорема Эккарта–Янга: A_k — наилучшее приближение

⚠ Теорема 0.3. Эккарт–Янг–Мирский (1936/1960)

Среди всех матриц B размера $m \times n$ ранга не более k матрица A_k из (0.4) даёт *наименьшую* ошибку в смысле спектральной нормы:

$$\min_{\operatorname{rank} B \le k} \| A - B \|_2 = \| A - A_k \|_2 = \sigma_{k+1}. \tag{3}$$

Здесь $\| M \|_2 = \max_{\|x\|=1} \| Mx \| = \sigma_1(M)$ — **спектральная норма** матрицы M (равна её наибольшему сингулярному числу). Утверждение остаётся в силе и для *фробениусовой* нормы $\| M \|_F = \sqrt{\sum_{i,j} m_{ij}^2}$:

$$\| A - A_k \|_F = \sqrt{\sigma_{k+1}^2 + \sigma_{k+2}^2 + \ldots + \sigma_r^2}. \tag{4}$$

Смысл теоремы: *если хочется сжать матрицу до ранга k, лучшее, что вы можете сделать, — оставить первые k компонент SVD.* Ошибка контролируется *отброшенным* сингулярным числом σ_{k+1}.

Пример: сжатие изображения

Возьмём фотографию 1024×768 в градациях серого. Хранение «в лоб» — это $1024 \cdot 768 = 786\,432$ чисел. Применим к матрице фото SVD и оставим только k старших компонент: придётся хранить

$$k \cdot (1024 + 768) + k = k \cdot 1793 \tag{5}$$

чисел — по одному столбцу u_i, одной строке $v_i^\top$ и одному числу σ_i на каждое слагаемое. Уже при $k = 50$ это $\sim 90\,000$ чисел — в *восемь* раз меньше исходного, а визуально потеря почти незаметна (рис. 0.3).

```
import numpy as np
import matplotlib.pyplot as plt
from skimage import data, color

# Load and convert to grayscale
img = color.rgb2gray(data.astronaut())      # shape (512, 512)
U, S, Vt = np.linalg.svd(img, full_matrices=False)

# Reconstruct using only top-k singular components
```

```
ranks = [1, 5, 20, 50, 200]
fig, axes = plt.subplots(1, len(ranks)+1, figsize=(14, 3))
axes[0].imshow(img, cmap='gray'); axes[0].set_title('original')
for ax, k in zip(axes[1:], ranks):
    A_k = U[:, :k] @ np.diag(S[:k]) @ Vt[:k, :]
    ax.imshow(A_k, cmap='gray')
    ax.set_title(f'rank {k}')
for ax in axes: ax.axis('off')
plt.tight_layout(); plt.show()
```

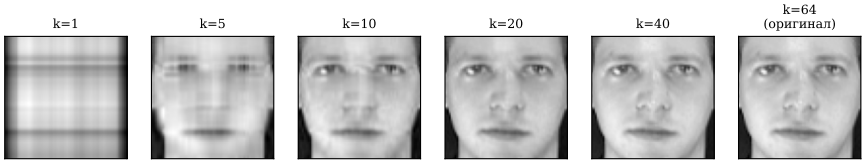

Рис. 0.3. Малоранговое приближение портрета астронавта. Уже при ранге 50 изображение визуально неотличимо от оригинала; при ранге 200 восстановление идеально. По теореме Эккарта–Янга это лучшее приближение фиксированного ранга.

i LoRA: малоранговая адаптация больших моделей

В 2021 г. Эдвард Ху с соавторами заметили: когда большую языковую модель дообучают на новую задачу, *обновление* матриц весов ΔW оказывается почти малоранговой матрицей. Поэтому вместо обучения «полной» ΔW можно сразу искать её в виде $\Delta W = U\bar{V}$ с маленьким k, где U, V — тонкие прямоугольники.

Это метод **LoRA** (*Low-Rank Adaptation*). Для модели с 175 миллиардами параметров такой приём сокращает объём дообучаемых весов в тысячи раз — и именно благодаря ему сегодня можно «дофайнтюнить» большую модель на бытовой видеокарте.

Теоретический фундамент LoRA — та самая теорема Эккарта–Янга 0.3, которой почти 90 лет.